MA501G Diskret matematik och logik, HT 2019

Datorövning 2

På datorövningarna kommer ni att använda Matlab för att lösa problem i diskret matematik och logik. Ni bör arbeta två och två, men det är tillåtet att jobba ensam om någon absolut föredrar detta. Ni ska inte vara tre eller fler i samma grupp.

På datorövningarna kommer ni att jobba på egen hand. En övningsledare kommer att finnas till hands och svara på frågor, men eftersom ni är många så kan ni ibland få vänta en stund på er tur att få hjälp. Sitter ni fast med en speciell uppgift, fråga de som sitter bredvid om de vet vad nästa steg är, eller gå vidare och titta på ett annat problem så länge. Det viktiga är att ni inte sitter långa stunder utan aktivitet.

Under datorövning 2 kommer ni att arbeta med aritmetik samt rekursiva definitioner och funktioner i Matlab. Titta i *Introduktion i programmering med MATLAB* från datorövning 1 om ni behöver repetera något. Jag påminner också om Matlabs funktion `help`; om ni skriver t.ex. `help ==` så berättar Matlab vad funktionen `==` gör.

Som en del av inlämning 4 ska ni redovisa era lösningar på 1a, 2ab, 3abc, 4ab och 5; gör 1b, 2c och 4c om ni får tid över.

1. (a) Skriv en rekursiv funktion `remainder` som beräknar den principala resten r då $n \geq 0$ divideras med $d > 0$. Testkör er funktion på uppgift 4 från inlämning 2.

   (b) Rekursiv programmering tar ofta mycket minne i anspråk. Skriv en funktion som beräknar r utan rekursion. Testkör enligt ovan. Tips: Använd Matlabs funktion `floor` och att $r = n - qd$, där q är kvoten.

2. Använd `remainder` för att lösa följande problem:

   (a) Skriv en funktion `prime` som avgör om ett heltal är ett primtal. Testkör er funktion.

   (b) Skriv en funktion `sgd` som använder Euklides algoritm för att beräkna den största gemensamma delaren av två positiva heltal. Testkör er funktion.

   (c) Skriv en funktion `klockaddition` som adderar två klockslag givna i timmar och minuter. T.ex. ska

   ```
   klockaddition([10 45],[14 30])
   ```

   ge svaret `[1 15]`.

3. På sidan 126 i kursboken introduceras Stirlingtal av andra ordningen. De betecknas $S(n,k)$ och har följande rekursiva definition:

$$\begin{cases} S(n,1) = S(n,n) = 1, \\ S(n,k) = S(n-1,k-1) + k \cdot S(n-1,k) & \text{om } 1 < k < n, \\ S(n,k) = 0 & \text{annars.} \end{cases}$$

   (a) Skriv en funktion `Stirling` som beräknar dessa tal.

   (b) Gör Övning 5.58 för hand.

   (c) Gör Övning 5.58 m.h.a. `Stirling`.

4. (a) Skriv en rekursiv funktion `fibo` som beräknar Fibonaccitalen.

   (b) Skriv en rekursiv funktion `fiboprod` som beräknar produkten $\prod_{i=k}^{n} f_i$, där $f_0, f_1, \ldots$ betecknar Fibonaccitalen.

   (c) Skriv en rekursiv funktion som gör samma sak som `fibo`, men fortare; se Övning 4.8.

5. Om man i Matlab skriver $[d, A, B] = \gcd(a, b)$, så blir $d = \gcd(a, b)$ medan A och B uppfyller $Aa + Bb = d$. Använd detta för att skriva en funktion `diofantos` som löser den diofantiska ekvationen $ax + by = c$. Testkör er funktion på Övning 3.26.